RankingSHAP – Faithful Listwise Feature Attribution Explanations for Ranking Models

Maria Heuss
University of Amsterdam, Amsterdam
The Netherlands
m.c.heuss@uva.nl

Maarten de Rijke
University of Amsterdam, Amsterdam
The Netherlands
m.derijke@uva.nl

Avishek Anand
Delft Institute of Technology, Delft
The Netherlands
Avishek.Anand@tudelft.nl

Abstract

While SHAP (SHapley Additive exPlanations) and other feature attribution methods are commonly employed to explain model predictions, their application within information retrieval (IR), particularly for complex outputs such as ranked lists, remains limited. Existing attribution methods typically provide pointwise explanations, focusing on why a single document received a high-ranking score, rather than considering the relationships between documents in a ranked list. We present three key contributions to address this gap. First, we rigorously define listwise feature attribution for ranking models. Secondly, we introduce RankingSHAP, extending the popular SHAP framework to accommodate listwise ranking attribution, addressing a significant methodological gap in the field. Third, we propose two novel evaluation paradigms for assessing the faithfulness of attributions in learning-to-rank models, measuring the correctness and completeness of the explanation with respect to different aspects. Through experiments on standard learning-to-rank datasets, we demonstrate RankingSHAP's practical application while identifying the constraints of selection-based explanations. We further employ a simulated study with an interpretable model to showcase how listwise ranking attributions can be used to examine model decisions and conduct a qualitative evaluation of explanations. Due to the contrastive nature of the ranking task, our understanding of ranking model decisions can substantially benefit from feature attribution explanations like RankingSHAP.

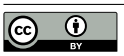
CCS Concepts

• **Information systems** → **Evaluation of retrieval results**.

Keywords

Explainable ranking systems, Explainability, Explanation evaluation, Feature attribution, Faithfulness

ACM Reference Format:
Maria Heuss, Maarten de Rijke, and Avishek Anand. 2025. RankingSHAP – Faithful Listwise Feature Attribution Explanations for Ranking Models. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3726302.3729971



SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3729971

1 Introduction

Feature attribution explanations are a posthoc family of explainability approaches that assign scores to features, quantifying their relative contribution to a model's decision. They are used to understand which features most influence the model's predictions, thereby enhancing transparency and trust. Feature attributions are among the most commonly used explanation types for posthoc explanations of trained models in general machine learning (ML) [20, 28, 35, 57].

Typical ML tasks involve pointwise prediction, explaining single classification or regression decisions. However, explaining rankings has different aspects – Why is a document relevant? (pointwise explanations), Why is one document more relevant than another? (pairwise explanations), or Why are the documents ranked in this specific order? (listwise explanations). Listwise explanations encode more context in terms of an entire or partial ranked list and are arguably more accurate/faithful since they are able to find features that affect an entire ranking. This is unlike feature attributions that focus on a single relevant document or a certain preference pair.

Feature attribution often lacks rigorous definition, beyond attributing the highest value to the *most important* feature. Limited work exists on pairwise [30] and listwise explanations [3, 25, 43, 44, 54]. Consequently, listwise feature attribution remains underexplored and in need of further theoretical underpinnings.

1.1 A Motivating Case Study – Talent Search

To motivate the need for tools that help practitioners arrive at a nuanced understanding of ranking outcomes, we consider talent search. There, systems use learning-to-rank to produce candidate rankings based on features like academic performance, experience, skills, and private attributes such as gender, ethnicity, and university attended. The inclusion of certain attributes in decision-making is debatable, as biases from past decisions can be reflected in the learned model and are best left to human judgment. However, sometimes these attributes are necessary for the model to perform well. Consider the two models in Fig. 1. Both use the same features, including skills, experience, graduation grade, university, and whether the candidate meets job requirements. The right model (Fig. 1b) uses the university reasonably by normalizing grades from different institutions, while the left model (Fig. 1a) discriminates against candidates from certain universities and favors others. Explanations can help differentiate between such models with similar performance to identify which is less biased and more trustworthy. Feature *selection* alone may not provide sufficient insights, as it likely selects the same features (x_{uni} and x_{rq}) for both models. Instead, feature *attribution*, which assigns each feature an importance value, can identify nuanced differences in their relative importance. Furthermore, since candidate ranking scores are only meaningful

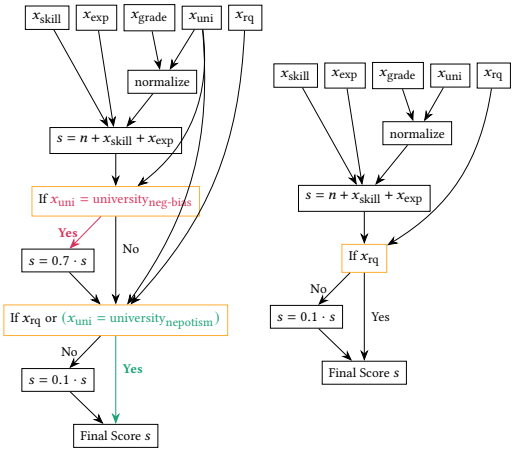


Figure 1: Flow chart of a biased and an unbiased model for a talent search task. With the help of explanations we would like to be able to differentiate between the two.

relative to others, *pointwise explanations* focusing on features for high scores may not reveal the university feature as the key factor in determining the relative order for queries with candidates from universities that the model is biased against. Pairwise and listwise explanations are better suited to explain relative rankings. While pairwise explanations require a specification of the pair of candidates to compare, listwise explanations can provide insight into the model decision as a whole. We will revisit this case study in Section 5 to demonstrate listwise feature attribution in practice.

1.2 Listwise Feature Attribution Explanations

We are interested in developing a listwise explanation method based on SHAP [24], a method inspired by Shapley values from *game theory*, that quantifies the contribution of each feature to a model's prediction. SHAP has gained significant popularity as a post-hoc explanation approach due to its theoretical properties and versatility [18]. However, SHAP only explains pointwise predictions: Given the contrastive nature of ranking tasks, listwise feature attribution would provide valuable insights into model decisions by explaining the relative order of documents, enabling comparisons across queries and ranking aspects. To address this gap, we introduce RankingSHAP, which extends SHAP to support listwise explanations while maintaining compatibility with existing research on SHAP's limitations and extensions. RankingSHAP provides flexibility in the *listwise explanation objective*, allowing users to determine feature importance for specific ranking aspects that *faithfully* reflect the model's behavior in the context of ranked lists.

1.3 Approach and Contributions

Our proposed method, RankingSHAP, preserves the context of ranked lists rather than evaluating documents in isolation. This contextual awareness is crucial because ranking models make decisions about relative document ordering. Therefore, a feature attribution method needs to identify a specific aspect of the model's decision to focus on and define a singular metric that quantifies changes within the ranked list with respect to that aspect. Aspects of interest may include a document's rank, measured by its shift in position, or the overall order of the top-k documents, measured by the number of permutations within the top-k. These diverse aspects underscore the need for a nuanced definition of listwise feature attribution in ranking models, which RankingSHAP provides.

We rigorously assess the faithfulness of RankingSHAP using established learning-to-rank (LtR) benchmark datasets, demonstrating its effectiveness in interpreting ranking models' outputs and providing deeper insights into their decision-making processes.

In summary, (i) we propose and rigorously define listwise feature attribution; (ii) we present a novel instantiation of our feature attribution framework called RankingSHAP; and (iii) we propose multiple evaluation schemes, white box check, preservation and deletion check for ranking feature attributions, and conduct extensive experiments to showcase RankingSHAP's performance.

2 Related Work

2.1 Shapley Values and SHAP

Shapley values, originating from game theory to define a player's marginal contribution [38], are widely used in explainable AI. Efficient approximation techniques have facilitated their application in AI model decisions [47, 48]. SHAP (SHapley Additive exPlanations) [24] is one such technique, approximating the expected marginal contribution of a feature to any feature set excluding it. A comprehensive overview and recent advancements are available in [28]; we build on this work, extending it for ranking models.

Contemporaneously with our work, Pliatsika et al. [31] propose a Shapley value-based framework for rankings and preferences, but our research emphasizes listwise explanations, unlike their document-level focus. Concurrently, Chowdhury et al. [7] establish theoretical properties for feature attribution in ranked lists and introduce a method similar to ours that satisfies these properties.

2.2 Explainable Information Retrieval

Explainable IR [3] has focused on models that are explainable by design [22, 56] and on approaches that can posthoc (after model training) explain models [42, 43, 50]. Posthoc approaches operate at the global level (model level) or at the local level (per-query). Global explainability approaches have been used to diagnose ad-hoc neural text rankers with well-understood axioms of text ranking [4, 34, 51] or to probe pre-trained transformer-based ranking models for ranking abilities [52]. We focus on *posthoc*, *local feature attributions*.

Feature Selection and Attribution for Ranking Models. Early work on interpreting ranking models was adapted for explaining query-document relevance from popular paradigms of black-box methods [24, 35] or white-box methods [39, 40, 49]. Singh and Anand [42], Verma and Ganguly [50] modify LIME [35], to generate terms as the explanation for a trained black-box ranker. Choi et al. [5], Fernando et al. [8] applied gradient-based feature attribution methods [24, 49] to interpret document relevance scores. Contrary to posthoc feature attribution approaches, local feature selection [12, 21, 22] approaches select a subset of features without distinguishing feature importance. Most work on local feature selection for rankings [12, 22] is not posthoc, and has been performed on text features, not on learning-to-rank data. In this work, we work on posthoc approaches for attribution and not selection.

Listwise Explanations for Ranking Models. Typical ML tasks are pointwise prediction tasks, i.e., focusing on a single classification or regression decision. In rankings, even for a single query, we also have to deal with pairwise and listwise explanations, which might be constructed by an aggregation of decisions. There has been limited work on pairwise [30] and listwise explanations [25, 37, 43, 54]. LiEGe [54] tackles the task as text generation. Other work uses simple rankers to approximate the original ranking of a complex black-box model by expanding query terms by solving a combinatorial optimization problem [25, 43]. The work that is closest work to ours, on RankLIME [6], approaches the problem with the local surrogate approach LIME, which the authors adapt for ranking models. Again, most of the approaches focus on text features and are not directly applicable to learning-to-rank models.

Explainability in Learning-to-Rank. Local feature selection approaches can be applied to learning-to-rank [9, 11, 32]. Among the feature-selection approaches, filter methods are model-agnostic [9], while wrapper methods are designed for a particular type of model [11]. In the context of ranking, some work produces local feature selections [32, 41]. Singh et al. [45] proposes the notions of validity and completeness based on the information contained in the explanation. While these notions are useful in both conception and evaluation of explanations, they still view the explanation as a *selection* of features. Feature selection methods, however, lack the capability to differentiate between features of varying importance, thereby avoiding a nuanced understanding of which features are substantially more critical in the decision-making process. We focus on feature attributions.

2.3 Faithfulness in Explainable AI

Faithfulness measures how accurately an explanation represents the reasoning process behind a model's prediction [13]. Evaluating faithfulness is challenging because the model's actual reasoning cannot be directly observed. Hence, various definitions and evaluation frameworks for faithfulness have been proposed [13, 26]. While there is no clear agreement as to what notion or framework should be used to measure and establish faithfulness [26], there are two dominant frameworks in explainable IR [3]. When locally approximating a ranking model with a proxy model, faithfulness is the degree to which the proxy model approximates the original ranking [25, 43]. An alternative notion of faithfulness is based on an *information-theoretic* notion of feature importance [45, 53]. There, faithfulness refers to the predictive power of the features in the attribution. Specifically, if a feature set is important then masking off or removing the non-relevant features should not result in a big change in model output. While both notions model different aspects of faithfulness, in this paper we follow the latter framework.

3 Feature Attribution for Pointwise Rankers

Early work on local feature explanations has introduced the concept of feature attribution [47]; recent work often lacks a clear definition of what makes a feature *important*, causing ambiguity in evaluating attribution faithfulness. Despite attempts to formalize feature attribution [2], these efforts have not been widely adopted, resulting in inconsistencies and confusion in the field [18]. We build on [24] to define *pointwise feature attribution* for black-box models with one-dimensional model output such as a pointwise ranking model

$$\tilde{R} : \mathcal{D} \to \mathbb{R}, x_{q,l} \mapsto s_{q,l}, \tag{1}$$

that predicts the ranking scores $s_{q,l} \in \mathbb{R}$, representing the probability of relevance, for the feature vectors of each document-query pair, $x_{q,l} \in \mathcal{D}$ in the space of all documents $\mathcal{D}$. We consider *instance-wise* feature attribution explanations that assign to each feature i an attribution value $\phi_i(x, \tilde{R})$, directly reflecting the importance of the feature to the model decision for instance x. Hence, *feature attribution explanations* can be understood as dictionaries $\{i \mapsto \phi_i(x, \tilde{R})\}_{i=1,\dots,n}$ containing exactly one attribution value per feature. A well-defined, instance-specific definition of feature attributions should consider the specific combinations of feature values in the input that collectively lead the model to predict a high score. Also, features with greater importance for the prediction should have higher attribution values.

We use marginal contributions to define *pointwise feature attribution*.[1] Our definition is based on SHAP [24]. In Section 4, we extend this to *listwise feature attribution* and define RankingSHAP to approximate feature attribution for listwise rankers.

Definition 3.1. We define the *attribution* or *importance* of a feature j in terms of marginal contributions. Let $n = dim(\mathcal{D})$ be the input space dimension, and let a *coalition* be a subset $S \subset \{1, \dots, n\} \setminus j$ of the input features excluding j. To measure the marginal contribution of feature j to coalition S, we compare the model output when shown only features in S to the output when shown features in $S \cup \{j\}$. Since we cannot simply erase features, we mask them with samples from a set of feature-vectors $B \subset \mathcal{D}$, called *background data*, which ideally summarizes the data distribution. For masking, we use templates defined by subsets S, indicating the presence ($i \in S$) or absence ($i \notin S$) of a feature, and data-points from the background data $b \in \mathcal{D}$. We define $m_{S,b} : \mathcal{D} \to \mathcal{D}$ as:

$$m_{S,b}(x)_i = \begin{cases} x_i, & \text{if } i \in S \\ b_i, & \text{if } i \notin S. \end{cases} \tag{2}$$

The marginal contribution of feature j to coalition S for vector b is:

$$\tilde{R}(m_{S\cup\{j\},b}(x)) - \tilde{R}(m_{S,b}(x)). \tag{3}$$

We define the **pointwise feature attribution** of feature j to the model decision of $\tilde{R}$ at input x as the expected marginal contribution of feature j to all possible coalitions of features:

$$\phi_j(x, \tilde{R}) = \sum_{S \subset \{1,\dots n\} \setminus j} w_S \cdot \mathbb{E}_{b\sim B}[\tilde{R}(m_{S\cup\{j\},b}(x)) - \tilde{R}(m_{S,b}(x))],$$

with weighting factor $w_S = \frac{1}{n!}|S|!(n - |S| - 1)!$ and uniform sampling from B.

Computational Costs. Given the exponential growth of coalitions with the number of features and the need for numerous background examples for a good summary, we approximate pointwise feature attribution using sampling. Following [24], we use SHAP for this approximation. Even though we are approximating the attribution values, SHAP is known to be computationally expensive, especially for high feature dimensions. There have been advances to making

[1] For a detailed discussion of marginal contributions, see [28].

the sampling more efficient [14, 55]. Also, since pointwise explanations are usually used as an analysis tool for specific input examples rather than to analyze the whole corpus, it remains a broadly used explanation approach [18, 28] despite its computational costs.

4 Feature Attribution for Listwise Rankers

For many machine learning tasks, SHapley Additive exPlanations (SHAP) [24] effectively approximate feature attribution values for individual model decisions, such as regression scores or classification probabilities. However, applying this method to listwise ranking models is challenging because these models output a ranked list rather than a single score. Within this ranked list, different decisions are made regarding the order of individual documents. Pointwise SHAP is only defined for a single one-dimensional model output. While it can explain the model score of an individual document, it does not consider the context of other documents in the list. In this work, we extend SHAP to an approach that caters to listwise ranking decisions, called RankingSHAP.

Instead of looking at pointwise ranking models, as we did in Section 3, we consider a listwise ranking model

$$R : \{\mathcal{D}_q\}_q \to \text{Sym}, \{x_{q,j}\}_j \mapsto \pi_q \tag{4}$$

that maps a set of candidate feature vectors for query q, $\mathcal{D}_q = \{x_{q,j}\}_j$, to some permutation matrix $\pi_q \in \text{Sym}(\mathcal{D}_q)$ representing the ranked list in the Symmetry group of all permutations of the candidate set $\mathcal{D}_q$.

We define two components, *listwise masking* and *listwise explanation objectives* that enable us to establish listwise feature attribution for ranking models, which we will introduce in Section 4.1. In Section 4.2, we formally define RankingSHAP for approximating listwise attribution values. We define RankingSHAP as a wrapper around SHAP using those two components. We deliberately chose not to modify SHAP's internal algorithm, allowing us to leverage the extensive literature on SHAP directly. Finally, we examine listwise explanation objectives with examples in Section 4.3.

4.1 Feature Attribution for Ranking Models

Our definition of feature attribution/feature importance for ranking models consists of two parts: (i) Define how masking applies to each document in the ranking $\mathcal{D}_q$ for query q. And (ii) measure the impact of input changes on the model decision, quantified by a single number.

Masking the Inputs of a Ranking Model. We apply a listwise mask $m_{S,b}$ to all documents $\{x_{q,j}\}_j$ in the ranking: $m_{S,b}(\mathcal{D}_q) = \prod_{x_{q,j} \in \mathcal{D}_q} m_{S,b}(x_{q,j})$. By masking the feature vector $x_{q,j}$ of each document with the same mask $m_{S,b}$, we disregard the impact of the masked features on the ranking decision. This helps identify the contributions of non-masked features to the document ordering.

Reducing the Model Prediction to a Single Prediction Value. Feature attribution is defined by the expected change in the predicted score. We need to reduce the ranking model's decisions to a single value reflecting the change for a perturbed input sample, using a listwise explanation objective that takes a ranked list and maps it to a value, highlighting some property of the ranked list that we want to investigate.

One example for such a function is a rank similarity coefficient like Kendall's tau τ [17], which is commonly used in the interpretability literature to measure rank correlation [25, 43, 45]. By comparing the change in the relative order of documents, we can measure how much the prediction deviates from the optimal order π_q predicted by the model:

$$g_q(\tilde{\pi}) = \tau(\pi_q, \tilde{\pi}), \tag{5}$$

For any such *listwise explanation objective* g_q, we define feature importance through the composition with the original ranking model, $g_q \circ R$. Section 4.3 provides further examples.

In summary, we have defined how to "remove" a feature from the model input through masking and measure its impact on the model prediction with a single value. This allows us to determine the **listwise feature attribution** using Section 3.

4.2 Estimating Listwise Feature Attribution with RankingSHAP

With the definition of feature attribution for ranking models, we introduce RankingSHAP. This depends on the choice of listwise explanation objective g and aims to explain which features are important for specific aspects of the ranked list. The ability to focus on different aspects of the ranking decision allows RankingSHAP to provide contrastive and flexible instance-wise explanations for rankers.

Following the definition of feature attribution with simultaneous masking of document vectors and a listwise explanation objective, we establish RankingSHAP as a wrapper around SHAP to approximate the marginal contribution of each feature in a ranking model, leveraging prior work.

SHAP samples both coalitions (templates for creating masks) and background data to generate masked perturbations (see Eq. 2) of the input, approximating the marginal contribution of a feature to any coalition. Given a sampled mask $m_{S,b}$, we illustrate how RankingSHAP adjusts the model prediction for use with SHAP in Algorithm 1. We loop over all documents $x_j \in \mathcal{D}_q$ (lines 1–2) and perturb the document features with the mask to get $\tilde{x}_j = m_{S,b}(x_j)$. Then, we rank the perturbed feature vectors with the ranking model $\pi = R(\{\tilde{x}_j\}_j)$ (line 3). Finally, we apply the listwise explanation objective $v = g(\pi)$ to measure the change in output according to the specified explanation objective (lines 4 and 5).

Computational Costs. Our approach allows for the use of existing SHAP implementations. This also means that it inherits any limitation that SHAP has such as the computational complexity. Nevertheless, it does not introduce any significant new additional computational overhead and allows us to use prior research on SHAP extensions and improvements for ranking without adjustments, such as advances in improving efficiency. Since SHAP is a commonly used explanation approach for pointwise predictions, we do not expect the computational complexity of RankingSHAP to hinder it's adoption in practice.

4.3 Listwise Explanation Objectives

We provide examples of listwise explanation objectives to illustrate the types of contrastive explanations RankingSHAP can generate.

Algorithm 1 Adjusted model prediction (used in combination with SHAP)

Require: ranking-model R, feature-vectors $\mathcal{D}_q$ for query q, listwise explanation objective g,
Input: masking function $m_{S,b}$
1: **for all** $x_j \in \mathcal{D}_q$ **do**
2: $\tilde{x}_j \leftarrow m_{S,b}(x_j)$
3: $\pi \leftarrow R(\{\tilde{x}_j\}_j)$
4: $v \leftarrow g(\pi)$
5: **return** v

Emphasizing Top-Ranked Documents. Instead of focusing on the entire ranked list, we can emphasize the top-k documents to identify features crucial for their high ranking. For example, we demonstrate RankingSHAP using a weighted rank difference objective with common position weighting:

$$g_q^w(\tilde{\pi}) = \sum_{d \in D_q} \frac{\text{rank}(d|\tilde{\pi}) - \text{rank}(d|\pi_q)}{\log_2(\text{rank}(d|\pi_q))}. \tag{6}$$

Explaining Feature Importance of a Singular Document. This objective focuses on one particular document d, investigating which features contribute, or would contribute, most to its high ranking compared to others when only a subset of features is considered. This can be implemented using the negative rank[2] of that document:

$$g_q^{\text{rank}(d)}(\tilde{\pi}) = -\text{rank}(d|\tilde{\pi}). \tag{7}$$

Alternatively, we can use RankingSHAP to determine the features that are the most beneficial for the document's exposure:

$$g_q^{\exp(d)}(\tilde{\pi}) = \exp(\text{rank}(d|\tilde{\pi})) = 1/\log_2(\text{rank}(d|\tilde{\pi})). \tag{8}$$

Explaining the Position of a Group of Documents. RankingSHAP allows us to compare ranking decisions for two groups of documents. We can consider the relative ordering or absolute distance of members of the different groups. Future work could explore explaining model fairness or identifying biases using listwise feature attribution.

5 Talent Search: A White Box Example

To demonstrate the application of RankingSHAP and to evaluate the feature attributes generated by different explanation approaches, we create a synthetic example, revisiting the talent search case study from the introduction. We design an interpretable model to estimate the importance of features for various model decisions. This evaluation framework, known as a "White Box Check," is widely used in the explainability community for other ML tasks [29].

In the following sections, we define features and ranking model that we will use as white box in Section 5.1. We then describe the experimental setup in Section 5.2 and examine various queries modeling different types of model decisions in Section 5.3. These queries demonstrate the practical use of listwise feature attribution and qualitatively evaluate three feature explanation approaches. In Section 5.4, we show how to use RankingSHAP to zoom in on individual documents and compare it to a pointwise explainer. We conclude with a detailed discussion in Section 5.5.

[2]We use the negative rank to maintain consistency with higher values being more desirable, explaining why a document ranks high (low rank) rather than low.

5.1 Model Design

We design a model using 5 features indicating whether a candidate meets general job requirements, the university the candidate graduated from, skill and experience levels, and average graduation grade. This model ranks candidates for various academic degree-required scenarios, aiming to mimic biases in trained models.

Table 1: Candidate evaluation criteria for running example

Feature	Description
Requirements	Binary value $x_{\text{rq}} \in \{\text{T}, \text{F}\}$ indicating if the candidate meets the job's minimum requirements.
Experience	Relevant work experience on a scale $x_{\text{exp}} \in [0, 1]$ (1=extensive experience, 0=none)
Skills	Skill fit on a scale $x_{\text{skill}} \in [0, 1]$, (1 = perfect match, 0 = no relevant skills)
University	Institution where the candidate obtained their degree, x_{uni}.
Grades	Mean graduation grade, x_{grade}, with range depending on the university.

Detailed feature information is in Table 1. The model favors candidates from $\textbf{uni}_{\textbf{nepotism}}$ and disadvantages those from $\textbf{uni}_{\textbf{neg-bias}}$. A flowchart is in Fig. 1a in Section 1. The ranking score is determined as follows:

- Normalize the grade $\textbf{norm}(x_{\text{grade}}, x_{\text{uni}})$, scaling it so that the minimum possible grade is 0 and the maximum is 1, to make grades from universities with different grading schemes comparable.
- Calculate the sum of x_{skill}, x_{exp}, and $\text{norm}(x_{\text{grade}}, x_{\text{uni}})$.
- For candidates from $\textbf{university}_{\textbf{neg-bias}}$, apply a negative bias by multiplying the score by 0.9.
- If the candidate does not meet the job requirements, multiply the score by 0.25, effectively placing them at the bottom of the list. Candidates from $\textbf{university}_{\textbf{nepotism}}$ are exempt from this penalty.

Candidates are ranked by their scores, with the highest at the top. We then investigate different queries with RankingSHAP to identify biases and compare attribution values to other explanation approaches.

5.2 Experimental Setup

The main goal of this Section is to showcase the usage of RankingSHAP and demonstrate the need for listwise, as opposed to pointwise, explanations and feature attribution rather than feature selection. Therefore, we compare RankingSHAP to the pointwise SHAP explainer, **PointwiseSHAP** (averaged over all candidates), as well as to the **Greedy** feature selection approach from [44]. The latter iteratively adds features to an initially empty set based on their marginal contribution to the Kendall's tau objective from Eq. 5 until the contribution becomes non-positive or the explanation size reaches 2. Section 6 contains a more complete empirical comparison with a comprehensive set of baselines, including RankLIME [6] and ShaRP [31]. For background data, we sample 100 candidates from uniform distributions over the possible feature values defined in Section 5.1. Detailed feature values and candidate lists for each query are provided in Appendix A.

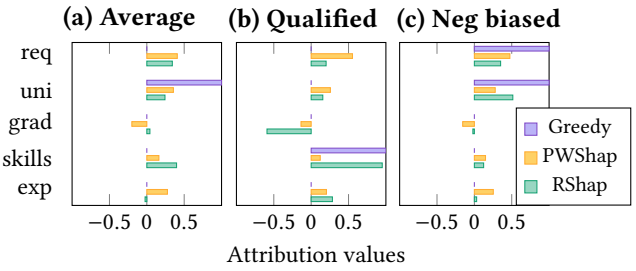


Figure 2: Feature attribution values for different query scenarios from Section 5.3.

5.3 Listwise Evaluation Across Query Scenarios

We define scenarios to demonstrate feature attribution for contrastive ranking explanations and evaluate them. We present 5 query scenarios: three in the main body and two in Appendix B.1.[3] We discuss the setup, candidate constellation, estimated feature importance $imp_{feature}$ for some features on the overall ranking, and evaluate the explanation approaches. In this part of our analysis RankingSHAP uses Kendall's tau explanation objective from Eq. 5 to explain the overall order of the candidates.

5.3.1 Average query. **Description.** This query includes candidates from universities with the same grading scheme, only some meeting the requirements, none from $\text{university}_{\text{neg-bias}}$ or $\text{university}_{\text{nepotism}}$.

Importance. Since no exceptions for candidates from biased institutes apply and grades are within the same scheme, we expect imp_{rq} to be high, as hiding this feature could change the ranking significantly. We also expect imp_{uni} to have a positive but smaller value since a change of university for all candidates causes ambiguity for the evaluation of the grade.

Evaluation of Feature Attributes. Fig. 2(a) shows attribution values/selected features (bars with length 1). Both RankingSHAP and PointwiseSHAP identify x_{rq} as an important feature and assign a positive value to x_{uni}. The greedy feature selection approach only selects the university feature.

5.3.2 Qualified query. **Description.** Similar to the average query, but only candidates meeting the requirements. The model can ignore x_{rq} without bias.

Importance. While imp_{uni} should still be assinged a positive value, imp_{rq} should be assigned a lower value than before as x_{rq} is irrelevant for these candidates.

Evaluation of the feature attributes. Fig. 2(b) shows that Greedy and RankingSHAP correctly assign a low value to the x_{rq}. PointwiseSHAP is not able to identify that the feature that is most important for attaining a high ranking score for each individual document, x_{rq}, is not important for this specific query. Furthermore, we notice that RankingSHAP assigns higher values to other features, that are now important to distinguish between the candidates.

5.3.3 Negative bias query. **Description.** Similar to the average query, with an additional candidate from $\text{university}_{\text{neg-bias}}$ having the best overall profile. The model has a negative bias towards this university.

Importance. We expect imp_{uni} to be higher due to the bias.

Evaluation of the feature attributes. In Fig. 2(c), both RankingSHAP and Greedy are able to identify the negative bias towards one candidate by correctly assigning a higher attribution value to x_{uni} than for the average query, while PointwiseSHAP is not.

5.4 Highlighting Feature Importance for the Rank of Individual Documents

In this section we zoom in on individual documents and the role of different features on the placement of that documents. For this analysis we use the exposure-based explanation objective from Eq. 8, highlighting the impact that the different features for the ranking model have on the exposure of the individual candidates. We compare to the attribution values generated by PointwiseSHAP for the specific document in question. We investigate two of the scenarios in more detail, the results for the other scenarios can be found in Appendix B.3. [4] Claims made in this subsection on the relative qualities of the candidates can be confirmed with Table 2 in Appendix A.

5.4.1 Qualified query. Since the university and requirements are the same for all candidates, a recruiter might be interested in which features were particularly important for ranking them. RankingSHAP provides more contrastive insight into the strengths of a document than PointwiseSHAP. For example, RankingSHAP highlights the skill feature as negatively impacting the third candidate's exposure. If a recruiter is more interested in grades, Fig. 3(a) allows them to make an informed decision to invite the candidate regardless of the model prediction. In contrast, PointwiseSHAP provides similar attribution values for each candidate and does not highlight the grades of the third-ranked candidate as a redeeming quality.

5.4.2 Biased query. The listwise feature attribution analysis of RankingSHAP from Fig. 2 shows high importance of the university feature for this query, warranting further investigation. Fig. 3(c) and (d) demonstrate that RankingSHAP can identify the unfair treatment of the third-ranked candidate due to their university, unlike PointwiseSHAP.

5.5 Discussion

The Contrastive Use of Feature Attribution. We define the estimated feature importance used in this section's evaluation in a contrastive way, comparing them to other queries as well as to other explanation objectives. Prior work [28] suggests that attribution values are hard to interpret in isolation; contextualizing them with other model decisions aids understanding. The use of different explanation objectives makes feature attribution particularly effective for ranking models: since a model decision involves a complex interplay of various decisions about the relative ordering of documents, contrasting different aspects of the decision allows us to uncover nuances that led to a specific model decision.

Using RankingSHAP to Identify Biases. By comparing attribution values of different queries, we can identify instances where a feature expected to be of moderate importance, such as x_{uni}, impacts the decision more than anticipated. For example, in the biased

[3] An extended appendix including these additional results is available at https://github.com/MariaHeuss/RankingShap/blob/main/Paper_RankingSHAP.pdf.

[4] An extended appendix including these additional results is available at https://github.com/MariaHeuss/RankingShap/blob/main/Paper_RankingSHAP.pdf.

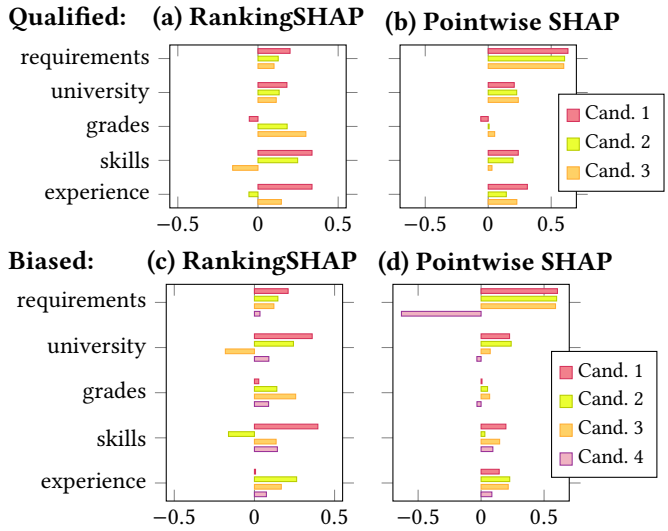


Figure 3: Feature attribution values, for RankingSHAP with the $g_q^{exp(d)}$ exposure objective defined in Section 4.3 and Pointwise SHAP for individual candidate in the ranked list.

query, we can detect hints of bias in the explanations in Section 5.3.3. Zooming in on what features are most important for the model to provide the individual candidates with exposure in Section 5.4, we see that RankingSHAP identifies the candidate that got negatively effected by the model bias, as well as qualities that might still speak for them.

Pointwise vs. Listwise Ranking Explanations. From our synthetic example we see that simply using a pointwise explanation approach to explain listwise ranking decisions fails to consider interactions between the feature values of different documents. Features that are important for a high ranking score are assigned a high attribution value, independent of whether they are important for the relative ordering of the list.

Selection is not Attribution. While feature selection can be a useful tool for understanding ranking models, more nuanced explanations are sometimes necessary to interpret model decisions. Even if the selection approach correctly identifies the most important features, a feature attribution approach is needed to gain detailed insight into the relative importance of the features impacting for example model bias.

Limitations of White Box Check Evaluation. We acknowledge the limitations of the qualitative evaluation in this section due to the subjective nature of estimated importance, the synthetic experiment setup, and the limited number of queries investigated. Nevertheless, this section is crucial for providing insights into using listwise feature attribution methods like RankingSHAP. To complement this qualitative evaluation, we will quantitatively compare RankingSHAP to a broad range of baselines in Section 6.

6 Quantitative Feature Attribution Evaluating

The quantitative evaluation of explanations is a difficult task [23]. In contrast to usual machine learning tasks, where labeled data to benchmark different models can be used for the evaluation, for explanations there is nothing like a *ground truth explanation*. Evaluating feature attribution values in particular is challenging, leading to prior work on evaluating feature attribution often defaults to evaluating the feature selection of the top-k features instead [36]. We will follow this strategy, by defining Preservation and Deletion Checks [29] for listwise explanations. We pose the following two research questions on the correctness/completeness of the explanations: (**RQ1**) Are explanations generated with RankingSHAP faithful to the model decision in terms of overall order of the documents? And (**RQ2**) Can RankingSHAP identify features responsible for the distribution of exposure in the ranked list? We describe our experimental setup in Section 6.1, our evaluation framework in Section 6.2, and our experimental results in Section 6.3.

6.1 Experimental Setup

6.1.1 Datasets. Following [44] we consider two datasets from LETOR4.0 [33]. MQ2008 consists of 800 queries with pre-computed query-document feature vectors of dimension 46. The MSLR data set consists of 10k queries with query-document feature vectors of dimension 136. For both, we use the train-val-test split of fold1 and evaluate the explanations on the test data.

6.1.2 Ranking model. We use the LightGBM [16] to train a listwise ranker with LambdaRank, using NDCG as metric.

6.1.3 Listwise Explanation Objectives. To provide additional evidence for the flexibility of RankingSHAP we use two different explanation objectives: **RShapK** uses Kendall's tau objective from Eq. 5 to identify features important for the overall ordering of candidate documents. **RShapW** employs the weighted rank difference objective g^w from Eq. 6 to prioritize documents ranked higher by the model.

6.1.4 Baselines. We consider the following baselines:
Random: Random feature attribution, normalized.
PWSHAP Previously used as a baseline in [44], we take the mean over the pointwise SHAP values of the top-5 documents.
PWLime: The mean over the pointwise attribution values generated with LIME of the top-5 documents.
Greedy: A greedy feature selection approach from [44]. The authors iteratively add features with the biggest marginal contribution to the initially empty explanation set until a set size of k is reached.
RLime: Listwise LIME for rankers, inspired by RankLIME [6]. Perturbation is done on each feature of each document independently. Since we are interested in listwise explanations, we report the mean of feature attribution values over all documents.
ShaRP As discussed in Section 2, parallel to our work, Pliatsika et al. [31] generate feature attribution explanations with SHAP for input features of individual documents, rather than the ranked list as a whole. We use the "Rank Quantity of Interest" for our implementation as it is closest in idea to our Kendall-tau based implementation of RankingSHAP. We use the mean of the individual document explanations to get listwise explanations.

6.1.5 Implementation details. All approaches, except Random, use background data for masking or perturbing input features. For MQ2008, we sample 100 random samples from the training data; for MSLR10k, we sample 20 to compensate for higher feature dimensions. For evaluation, we sample a different set of 100 background samples for both datasets. We use the KernelSHAP implementation

from the SHAP library [24] for RankingSHAP, PWShap and ShaRP and the TabularExplainer from the LIME library [35] for PWLime and RLime, all with default settings.

6.2 Experimental Evaluation

Due to the lack of ground truth attribution values and evaluation frameworks for rankers, we use the deletion and preservation check strategy [29] from other machine learning tasks, adapted for ranking. A good explanation should replicate the original model output when non-explained features are masked (Preservation check) and significantly alter the output when important features are removed (Deletion check).

Both checks measure the impact of masking features on the model output, evaluated by a function v. We sample masking values b from background data B to substitute for non-explained features, resulting in re-ranked lists $\tilde{\pi}_{e,b}$:

$$\text{Preservation(e)} = \mathbb{E}_{\text{b}\sim\text{B}}[\text{v}(\tilde{\pi}_{\text{e,b}})] \,.$$

Similarly, the deletion check applies the mask to the features included in the explanation.

For ranked list outputs, we use Kendall's similarity τ with the original ranked list π, hence $v^{\tau}(\tilde{\pi}_{e,b}) = \tau(\pi, \tilde{\pi}_{e,b})$. These checks align with the validity and completeness criteria in [44]. Additionally, we evaluate the alignment of the generated explanations with the original model by measuring the exposure difference between each candidate ranked with the original input and the masked input: $v^{\text{exp-diff}}(\tilde{\pi}_{e,b}) = \sum_{d\in\pi} |\exp(\text{rank}(d|\pi)) - \exp(\text{rank}(d|\tilde{\pi}_{e,b}))|$. We conduct evaluations at explanation sizes of 1, 3, 5, 7, and 10 and report the mean values over all evaluated queries.

Note that in this approach, we evaluate feature selection explanations as subsets of features, not attribution values. For feature attribution explanations, we use the top-k features.

6.3 Results

The results with the deletion and preservation checks are presented in Fig. 4.

(RQ1) Are Explanations Generated with RankingSHAP Faithful to the Model Decision in Terms of Overall Order of the Documents. To address this research question, we evaluate the *correctness* (how well the explanation aligns with the model's decision) and *completeness* (how much relevant information is captured in the features with the highest attribution values) of the explanations. The preservation check with rank-similarity measures how well the ranked list can be reconstructed using only the most important features identified by each explanation approach. As shown in Fig. 4 (a), only the Greedy baseline outperforms RankingSHAP, which is expected since Greedy is designed to maximize this metric through feature selection explanations. Conversely, the deletion check (b), which involves removing the features with the highest attribution values, reveals that RankingSHAP outperforms all baselines, including the Greedy and all pointwise baselines. These findings are consistent for the MSLR dataset, as illustrated in Fig. 4 (c) and (d). Overall using an explanation size of 10 features, we achieve approximately 0.7 rank similarity for the MQ2008 data and 0.6 rank similarity for the MSLR-10k data. In contrast, the rank similarity drops to less than 0.2 and 0.4, respectively, when removing these 10 features with the highest attribution values from the model input. Thus, we answer our first research question in the affirmative: RankingSHAP is capable of faithfully explaining the model decision.

(RQ2) Can RankingSHAP Identify Features Responsible for the Distribution of Exposure in the Ranked List. We compare explanation approaches using the Preservation Check (Fig. 4 (e) and (g)) and the Deletion Check (Fig. 4 (f) and (h)), alongside the exposure difference metric $v^{\text{exp-diff}}$ from Section 6.2. The Preservation Check indicates that the exposure difference decreases for all explanation approaches as the explanation size increases. RankingSHAP and the Greedy approach perform best in the Preservation Check, reducing the exposure difference by 1/2 to 1/3 compared to the random baseline. In the Deletion Check, RankingSHAP clearly outperforms all other approaches, producing an exposure difference 3 to 5 times greater, depending on the dataset, when the most important features identified by RankingSHAP are omitted, as opposed to random features. These findings provide evidence that RankingSHAP effectively identifies features responsible for the distribution of exposure in a ranked list, thus positively answering the research question.

6.4 Reflections

Using Different Explanation Objectives for Focusing on Different Aspects of the Ranking Decision. The performance difference between the two versions of RankingSHAP, each with distinct explanation objectives, highlights RankingSHAP's ability to emphasize different aspects of the ranked list for specialized explanations. A listwise similarity objective, like Kendall's tau in RShapK, identifies features critical for the overall ranking. Conversely, an objective like the weighted rank difference in WShapK focuses on the top of the ranked list, improving faithfulness for top documents, as evidenced by exposure-based evaluation. Hence, when using RankingSHAP for generating ranking explanations, it is crucial to carefully consider which aspects of the ranking decision should be elucidated.

Using SHAP Advances in RankingSHAP for Enhanced Interpretability. Since we define RankingSHAP as a wrapper around SHAP, it is possible to apply improvements developed for SHAP to RankingSHAP. This allows for the use of numerous advances in the field, such as handling correlated features [1], increasing the efficiency of SHAP [14, 15], and making adjustments to the sampling of background data [10], or the weighting of different coalitions when calculating SHAP values [20]. Some of these advances can be applied directly to RankingSHAP, although future research will need to investigate how easily transferable these improvements are to the ranking task.

7 Conclusion

In this work, we have defined the concept of listwise feature attribution for ranking tasks, allowing flexible and contrastive examination of ranking decisions through a listwise explanation objective. We show that our proposed approach RankingSHAP results in delivering faithful feature attributions and RankingSHAP can aid in meaningfully understanding model decisions and detecting biases.

However, we note that RankingSHAP has limitations, including high computational costs for high-dimensional input spaces and

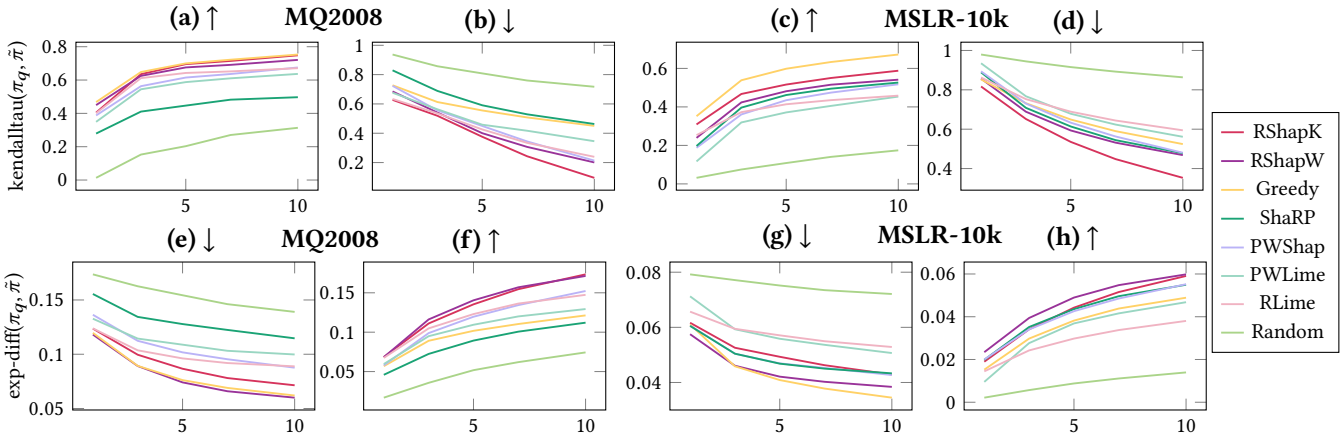


Figure 4: Preservation (a, c, e, g) and Deletion Check (b, d, f, h). Only features top-k of the explanations are kept/ masked. For the kendalltau measure, higher numbers represent higher similarity with the original rank, so for the Preservation check higher is better while for the Deletion check lower is better. For the exposure-base measure, it is exactly the other way around since lower numbers represent exposure closer to the original one.

the challenge of interpreting SHAP values, which may not always align with human expectations [19], potentially lacking contrastiveness [27], and it can be susceptible to adversarial attacks [46]. Additionally, SHAP assumes uncorrelated features, leading to unrealistic out-of-distribution data if ignored [1]. Some of these limitations have been addressed in prior literature, and due to RankingSHAP's structure as a SHAP wrapper, these improvements could potentially be applied to RankingSHAP (see Section 6.4).

For future work, we see the need for a more thorough evaluation framework that goes beyond faithfulness. Furthermore, future research should examine whether using listwise SHAP attribution values in a contrastive manner can bridge the gap between mathematically well-defined explanations and practical applications in real-life scenarios.

Data and code. To facilitate reproducibility, code and parameters are available at https://github.com/MariaHeuss/RankingShap.

Acknowledgments

We thank Joeri Noorthoek for contributions to the code used in this work, Philipp Hager for feedback on the code, Mathijs Henquet and Jasmin Kareem for feedback on the manuscript. This research was (partially) supported by the Dutch Research Council (NWO), under project numbers 024.004.022, NWA.1389.20.183, and KICH3.LTP.20.006, the European Union's Horizon Europe program under grant agreement No 101070212, the German Research Foundation (DFG), under Project IREM with grant No. AN 996/1-1. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

A Appendix

Here we include the explicit set-up of the simulated example from Section 5. In Table 2 we give an overview over all candidates that were used for the different query scenarios. The different universities have different grading schemes, which the models from Figure 1 depends on. Table 3 shows an overview over the different universities that are used in the query scenarios. We show the best possible

Table 2: Feature values for the individual candidates.

candidate	experience	skills	grades	university	req
qual-1	0.8	0.55	3.5	uni_{us}	True
qual-2	0.7	0.75	3.3	uni_{us}	True
qual-3	0.9	0.8	3	uni_{us}	True
non-qual	0.7	0.7	3	uni_{us}	False
privileged	0.8	0.6	3.6	uni_{nep}	False
qual-net	0.7	0.9	8	uni_{net}	True
qual-ger	0.8	0.8	1	uni_{ger}	True
qual-biased	0.8	0.7	3.6	uni_{bias}	True

and the worst passing grade as well as whether the biased model is biased towards the university in question. Those candidates were

Table 3: Comparison of grading schemes and model bias across universities.

university	highest grade	lowest grade	model bias
uni_{us}	4	1	None
uni_{nep}	4	1	Positive
uni_{bias}	4	1	Negative
uni_{ger}	1	4	None
uni_{net}	10	6	None

then used for different queries. Which candidates were used for what queries can be found in Table 4. The table entries indicate the rank of the candidate for the biased ranker, with 0 indicating that they were not included.

Table 4: Query-candidate matrix - numbers indicate the rank for the biased ranker, 0 that they were not considered.

candidate	average	nepotism	qualified	internat.	biased
qual-1	2	3	3	0	3
qual-2	1	2	2	0	2
qual-3	0	0	1	2	0
non-qual	3	4	0	4	4
privileged	0	1	0	0	0
qual-net	0	0	0	3	0
qual-ger	0	0	0	1	0
qual-biased	0	0	0	0	1

References

[1] Kjersti Aas, Martin Jullum, and Anders Løland. 2021. Explaining Individual Predictions When Features are Dependent: More Accurate Approximations to Shapley Values. *Artificial Intelligence* 298 (2021), 103502.

[2] Darius Afchar, Vincent Guigue, and Romain Hennequin. 2021. Towards Rigorous Interpretations: A Formalisation of Feature Attribution. In *International Conference on Machine Learning*. PMLR, 76–86.

[3] Avishek Anand, Procheta Sen, Sourav Saha, Manisha Verma, and Mandar Mitra. 2023. Explainable Information Retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3448–3451.

[4] Arthur Câmara and Claudia Hauff. 2020. Diagnosing BERT with Retrieval Heuristics. *Advances in Information Retrieval* 12035 (2020), 605.

[5] Jaekeol Choi, Jungin Choi, and Wonjong Rhee. 2020. Interpreting Neural Ranking Models Using Grad-CAM. *arXiv preprint arXiv:2005.05768* (2020).

[6] Tanya Chowdhury, Razieh Rahimi, and James Allan. 2023. Rank-LIME: Local Model-agnostic Feature Attribution for Learning to Rank. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval*. 33–37.

[7] Tanya Chowdhury, Yair Zick, and James Allan. 2024. RankSHAP: a Gold Standard Feature Attribution Method for the Ranking Task. *arXiv preprint arXiv:2405.01848* (2024).

[8] Zeon Trevor Fernando, Jaspreet Singh, and Avishek Anand. 2019. A Study on the Interpretability of Neural Retrieval Models Using DeepSHAP. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. ACM, New York, NY, USA, 1005–1008.

[9] Xiubo Geng, Tie-Yan Liu, Tao Qin, and Hang Li. 2007. Feature Selection for Ranking. In *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007*, Wessel Kraaij, Arjen P. de Vries, Charles L. A. Clarke, Norbert Fuhr, and Noriko Kando (Eds.). ACM, 407–414.

[10] Sahra Ghalebikesabi, Lucile Ter-Minassian, Karla DiazOrdaz, and Chris C Holmes. 2021. On Locality of Local Explanation Models. *Advances in neural information processing systems* 34 (2021), 18395–18407.

[11] Andrea Gigli, Claudio Lucchese, Franco Maria Nardini, and Raffaele Perego. 2016. Fast Feature Selection for Learning to Rank. In *Proceedings of the 2016 ACM on International Conference on the Theory of Information Retrieval, ICTIR 2016, Newark, DE, USA, September 12- 6, 2016*, Ben Carterette, Hui Fang, Mounia Lalmas, and Jian-Yun Nie (Eds.). ACM, 167–170.

[12] Sebastian Hofstätter, Bhaskar Mitra, Hamed Zamani, Nick Craswell, and Allan Hanbury. 2021. Intra-Document Cascading: Learning to Select Passages for Neural Document Ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 1349–1358.

[13] Alon Jacovi and Yoav Goldberg. 2020. Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness?. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 4198–4205.

[14] Neil Jethani, Mukund Sudarshan, Ian Connick Covert, Su-In Lee, and Rajesh Ranganath. 2021. FastSHAP: Real-time Shapley Value Estimation. In *International conference on learning representations*.

[15] Sanjay Kariyappa, Leonidas Tsepenekas, Freddy Lécué, and Daniele Magazzeni. 2024. SHAP@ k: Efficient and Probably Approximately Correct (PAC) Identification of Top-k Features. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 13068–13075.

[16] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in neural information processing systems*, Vol. 30.

[17] Maurice G. Kendall. 1938. A New Measure of Rank Correlation. *Biometrika* 30, 1/2 (1938), 81–93.

[18] Satyapriya Krishna, Tessa Han, Alex Gu, Javin Pombra, Shahin Jabbari, Steven Wu, and Himabindu Lakkaraju. 2022. The Disagreement Problem in Explainable Machine Learning: A Practitioner's Perspective. *arXiv preprint arXiv:2202.01602* (2022).

[19] I. Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle Friedler. 2020. Problems with Shapley-value-based Explanations as Feature Importance Measures. In *International Conference on Machine Learning*. PMLR, 5491–5500.

[20] Yongchan Kwon and James Y Zou. 2022. WeightedSHAP: Analyzing and Improving Shapley based Feature Attributions. *Advances in Neural Information Processing Systems* 35 (2022), 34363–34376.

[21] Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing Neural Predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 107–117.

[22] Jurek Leonhardt, Koustav Rudra, and Avishek Anand. 2021. Learnt Sparsity for Effective and Interpretable Document Ranking. *arXiv preprint arXiv:2106.12460* (2021).

[23] Ana Lucic, Madhulika Srikumar, Umang Bhatt, Alice Xiang, Ankur Taly, Q. Vera Liao, and Maarten de Rijke. 2021. A Multistakeholder Approach Towards Evaluating AI Transparency Mechanisms. In *ACM CHI Workshop on Operationalizing Human-Centered Perspectives in Explainable AI*. ACM.

[24] Scott M. Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems* 30 (2017).

[25] Lijun Lyu and Avishek Anand. 2023. Listwise Explanations for Ranking Models Using Multiple Explainers. In *European Conference on Information Retrieval*. Springer Nature Switzerland Cham, 653–668.

[26] Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2024. Towards faithful model explanation in nlp: A survey. *Computational Linguistics* (2024), 1–67.

[27] Tim Miller. 2019. Explanation in Artificial Intelligence: Insights from the Social Sciences. *Artificial intelligence* 267 (2019), 1–38.

[28] Christophe Molnar. 2023. *Interpreting Machine Learning Models with SHAP*. Independently published.

[29] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. 2023. From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI. *Comput. Surveys* 55, 13s (2023), 1–42.

[30] Gustavo Penha, Eyal Krikon, and Vanessa Murdock. 2022. Pairwise Review-based Explanations for Voice Product Search. In *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*. 300–304.

[31] Venetia Pliatsika, Joao Fonseca, Tilun Wang, and Julia Stoyanovich. 2024. ShaRP: Explaining Rankings with Shapley Values. *arXiv preprint arXiv:2401.16744* (2024).

[32] Alberto Purpura, Karolina Buchner, Gianmaria Silvello, and Gian Antonio Susto. 2021. Neural Feature Selection for Learning to Rank. In *European Conference on Information Retrieval*. Springer, 342–349.

[33] Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 Datasets. *arXiv preprint arXiv:1306.2597* (2013).

[34] Daniël Rennings, Felipe Moraes, and Claudia Hauff. 2019. An Axiomatic Approach to Diagnosing Neural IR Models. In *European Conference on Information Retrieval*. Springer, 489–503.

[35] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1135–1144.

[36] Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. 2022. A Consistent and Efficient Evaluation Strategy for Attribution Methods. In *Proceedings of the 39th International Conference on Machine Learning*. 18770–18795.

[37] Sourav Saha, Harsh Agarwal, Swastik Mohanty, Mandar Mitra, and Debapriyo Majumdar. 2024. ir_explain: a Python Library of Explainable IR Methods. *arXiv preprint arXiv:2404.18546* (2024).

[38] Lloyd S. Shapley. 1953. A Value for n-Person Games. In *Contributions to the Theory of Games II*. Princeton University Press, Princeton, 307–317.

[39] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning Important Features Through Propagating Activation Differences. In *International Conference on Machine Learning*. PMLR, 3145–3153.

[40] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *2nd International Conference on Learning Representations, ICLR 2014, Workshop Track Proceedings*.

[41] Jaspreet Singh and Avishek Anand. 2018. Posthoc Interpretability of Learning to Rank Models using Secondary Training Data. *arXiv preprint arXiv:1806.11330* (2018).

[42] Jaspreet Singh and Avishek Anand. 2019. EXS: Explainable Search Using Local Model Agnostic Interpretability. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining* (Melbourne VIC, Australia) *(WSDM '19)*. ACM, New York, NY, USA, 770–773.

[43] Jaspreet Singh and Avishek Anand. 2020. Model Agnostic Interpretability of Rankers via Intent Modelling. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 618–628.

[44] Jaspreet Singh, Megha Khosla, Wang Zhenye, and Avishek Anand. 2021. Extracting per Query Valid Explanations for Blackbox Learning-to-Rank Models. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*. 203–210.

[45] Jaspreet Singh, Zhenye Wang, Megha Khosla, and Avishek Anand. 2021. Valid Explanations for Learning to Rank Models. *International Conference on the Theory of Information Retrieval* (2021).

[46] Dylan Slack, Sophie Hilgard, Emily Jia, Sameer Singh, and Himabindu Lakkaraju. 2020. Fooling Lime and SHAP: Adversarial Attacks on Post Hoc Explanation Methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 180–186.

[47] Erik Strumbelj and Igor Kononenko. 2010. An Efficient Explanation of Individual Classifications using Game Theory. *The Journal of Machine Learning Research* 11 (2010), 1–18.

[48] Erik Štrumbelj and Igor Kononenko. 2014. Explaining Prediction Models and Individual Predictions with Feature Contributions. *Knowledge and information*

systems 41 (2014), 647–665.

[49] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. In *International Conference on Machine Learning*. PMLR, 3319–3328.

[50] Manisha Verma and Debasis Ganguly. 2019. LIRME: Locally Interpretable Ranking Model Explanation. In *Proceedings of the 42nd International ACM SIGIR*.

[51] Michael Völske, Alexander Bondarenko, Maik Fröbe, Matthias Hagen, Benno Stein, Jaspreet Singh, and Avishek Anand. 2021. Towards Axiomatic Explanations for Neural Ranking Models. *International Conference on the Theory of Information Retrieval* (2021).

[52] Jonas Wallat, Fabian Beringer, Abhijit Anand, and Avishek Anand. 2023. Probing BERT for Ranking Abilities. In *European Conference on Information Retrieval*. Springer Nature Switzerland Cham, 255–273.

[53] Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not Explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 11–20.

[54] Puxuan Yu, Razieh Rahimi, and James Allan. 2022. Towards Explainable Search Results: A Listwise Explanation Generator. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 669–680.

[55] Jiayao Zhang, Qiheng Sun, Jinfei Liu, Li Xiong, Jian Pei, and Kui Ren. 2023. Efficient sampling approaches to shapley value approximation. *Proceedings of the ACM on Management of Data* 1, 1 (2023), 1–24.

[56] Zijian Zhang, Koustav Rudra, and Avishek Anand. 2021. Explain and Predict, and then Predict again. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 418–426.

[57] Jianlong Zhou, Amir H. Gandomi, Fang Chen, and Andreas Holzinger. 2021. Evaluating the Quality of Machine Learning Explanations: A Survey on Methods and Metrics. *Electronics* 10, 5 (2021), 593.

B Simulated experiment - additional results.

Here we present additional results for the simulated experiment for some more query scenarios, as well as for the unbiased model from the flowchart in Figure 1b.

B.1 Additional query scenarios

B.1.1 Nepotism query. **Description.** For this query, one additional candidate from $\text{university}_{\text{nepotism}}$ with good records for x_{skill}, x_{exp} and x_{grade} is considered, but lacking some of the job requirements.

Importance. As we know, the model has picked up on a bias in the data, favoring candidates coming from $\text{university}_{\text{nepotism}}$, which coincidentally or not is the same university that some people that made past hiring decisions graduated from. Hence, for this query we estimate imp_{rq} to take a smaller value, and imp_{uni} to take a higher importance value.

Evaluation of the feature attributes. In Fig. 5(b) we see that all approaches correctly pick up on the bias towards $\text{university}_{\text{nepotism}}$ by assigning a high value to x_{uni}, while assigning a low value to/ not selecting the usually important x_{req}.

B.1.2 International query. **Description.**This query considers candidates from universities with different grading schemes. Most candidates meet the job requirements, and none are from $\text{university}_{\text{nepotism}}$ or $\text{university}_{\text{neg-bias}}$

Importance. For this query we estimate imp_{uni} to take a higher value than for the average query. Since candidates from universities with different grading schemes are compared, knowing which university the candidate went to is important for the interpretation of the grades.

Evaluation of the feature attributes. By comparing Fig. 5(d), with the plot for the average query (a) we see that RankingSHAP is the only approach assigning x_{uni} a higher value than for the average query.

B.2 Unbiased model explanations

The bar chart in Figure 6 shows the feature attribution values from the three considered approaches from Section 5.2 for the same query scenarios as defined in Section 5.3. Comparing the attribution values of different models for different query scenarios like in Figures 5 and 6 can help us with selecting the least biased model when we have a choice of models of similar performance.

B.3 Additional per candidate analysis

Here we provide additional results of the per candidate analysis from Section 5.4.

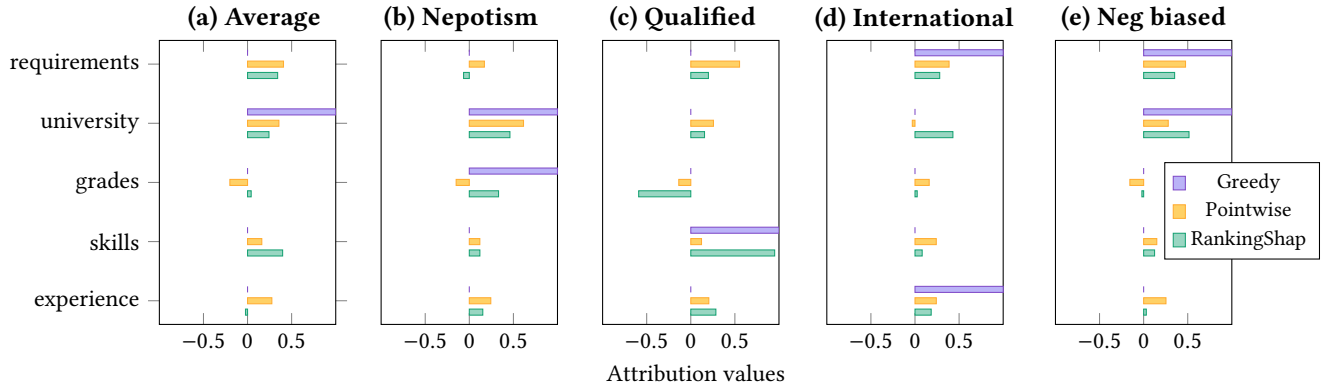


Figure 5: Feature attribution values of the biased model for different query scenarios from Section 5.3 including two additional queries

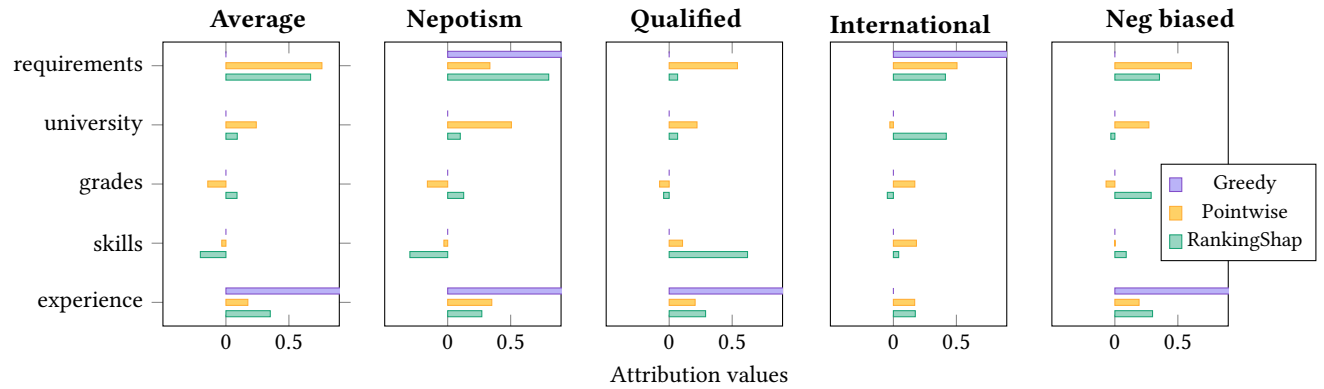


Figure 6: Feature attribution values of the unbiased model for different query scenarios

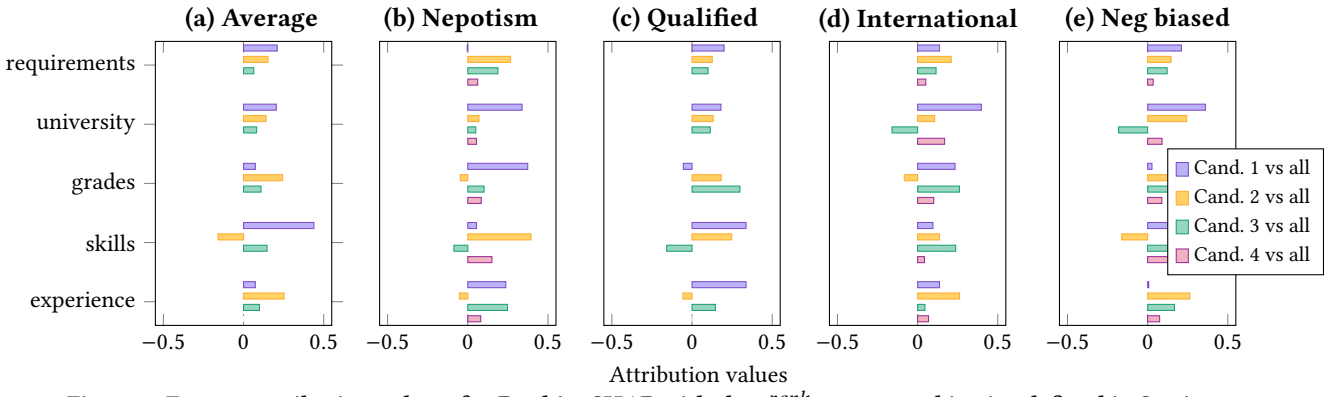


Figure 7: Feature attribution values, for RankingSHAP with the g_q^{rank} exposure objective defined in Section 4.3

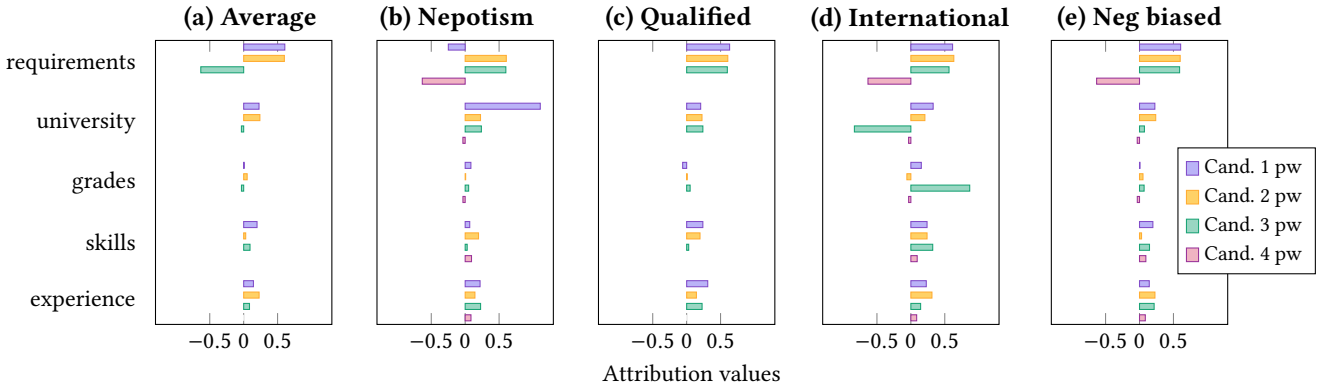


Figure 8: Feature attribution values, Pointwise SHAP for each individual candidate in the ranked list.